What Is Claimed Is:

1   1.   A method for checkpointing an application, comprising:

2   pre-linking an interceptor library into the application during a run-time

3 invocation of the application, wherein the run-time invocation occurs after the

4 application has been complied and linked;

5   intercepting a function call produced by the application at the interceptor

6 library;

7   recording parameters of the function call to create a checkpoint that

8 includes information about the function call parameters;

9   making the function call;

10   receiving results of the function call; and

11   forwarding results of the function call back to the application.

1   2.   The method of claim 1, further comprising creating a checkpoint

2 by:

3   stopping the application;

4   retrieving the recorded parameters;

5   saving the checkpoint data, including the recorded parameters, to

6 secondary storage; and

7   resuming the application.

1   3.   The method of claim 2, further comprising using the checkpoint to

2 restore the application.

1   4.   The method of claim 2, wherein saving the checkpoint data to

2 secondary storage involves saving the checkpoint data to a persistent storage.

9

1   5.   The method of claim 2, wherein saving the checkpoint data to

2 secondary storage involves saving the checkpoint data in a file system, or a

3 database.

1   6.   The method of claim 1, wherein making the function call involves

2 referencing the function through a function pointer.

1   7.   The method of claim 1, further comprising recording the results of

2 the function call to facilitate creating a checkpoint that includes information about

3 the results of the function call.

1   8.   The method of claim 1, wherein the function calls can include

2 system calls or lib calls.

1   9.   The method of claim 1, wherein the parameters can include:

2   file paths;

3   thread flags; and

4   timer-thread relationships.

1   10.   A computer-readable storage medium storing instructions that

2 when executed by a computer cause the computer to perform a method for

3 checkpointing an application, the method comprising:

4   pre-linking an interceptor library into the application during a run-time

5 invocation of the application, wherein the run-time invocation occurs after the

6 application has been complied and linked;

10

7   intercepting a function call produced by the application at the interceptor

8   library;

9   recording parameters of the function call to create a checkpoint that

10   includes information about the function call parameters;

11   making the function call;

12   receiving results of the function call; and

13   forwarding results of the function call back to the application.

1   11.   The computer-readable storage medium of claim 10, further

2   comprising creating a checkpoint by:

3   stopping the application;

4   retrieving the recorded parameters;

5   saving the checkpoint data, including the recorded parameters, to

6   secondary storage; and

7   resuming the application.

1   12.   The computer-readable storage medium of claim 11, further

2   comprising using the checkpoint to restore the application.

1   13.   The computer-readable storage medium of claim 11, wherein

2   saving the checkpoint data to secondary storage involves saving the checkpoint

3   data to a persistent storage.

1   14.   The computer-readable storage medium of claim 12, wherein

2   saving the checkpoint data to secondary storage involves saving the checkpoint

3   data in a file system, or a database.

11

Attorney Docket No. SUN-P6316-RSH   Inventor(s): Mathiske, et al.

ARPH \SUN MICROSYSTEMS\SUN-P6316-RSH\SUN-P6316-RSH APPLICATION DOC

1   15.   The computer-readable storage medium of claim 10, wherein

2   making the function call involves referencing the function through a function

3   pointer.

1   16.   The computer-readable storage medium of claim 10, wherein the

2   method further comprises recording the results of the function call to facilitate

3   creating a checkpoint that includes information about the results of the function

4   call.

1   17.   The computer-readable storage medium of claim 10, wherein the

2   function calls can include system calls or lib calls.

1   18.   The computer-readable storage medium of claim 10, wherein the

2   parameters can include:

3   file paths;

4   thread flags; and

5   timer-thread relationships.

1   19.   An apparatus that checkpoints an application, comprising:

2   a pre-linking mechanism that is configured to pre-link an interceptor

3   library into the application during a run-time invocation of the application,

4   wherein the run-time invocation occurs after the application has been complied

5   and linked;

6   an intercepting mechanism within the interceptor library that is configured

7   to intercept a function call produced by the application;

8   a recording mechanism that is configured to record parameters of the

9   function call to facilitate creating a checkpoint that includes information about the

12

10   function call parameters;

11   a calling mechanism that is configured to make the function call;

12   a receiving mechanism that is configured to receive results of the function

13   call; and

14   a forwarding mechanism that is configured to forward results of the

15   function call back to the application.

1   20.   The apparatus of claim 19, further comprising a checkpoint

2   creation mechanism that is configured to:

3   stop the application;

4   retrieve the recorded parameters;

5   save the checkpoint data, including the recorded parameters, to secondary

6   storage; and to

7   resume the application.

1   21.   The apparatus of claim 20, further comprising a restoration

2   mechanism that is configured to use the checkpoint data to restore the application

3   to the checkpointed state.

1   22.   The apparatus of claim 20, wherein the checkpoint creation

2   mechanism is configured to save checkpoint data to a persistent storage.

1   23.   The apparatus of claim 20, wherein the checkpoint creation

2   mechanism is configured to save the checkpoint data in a file system, or a

3   database.

1   24.   The apparatus of claim 19, wherein the calling mechanism is

13

2   configured to make the function call by referencing the function through a

3   function pointer.

1   25.   The apparatus of claim 19, further comprising a recording

2   mechanism that is configured to record the results of the function call to facilitate

3   creating a checkpoint that includes information about the results of the function

4   call.

1   26.   The apparatus of claim 19, wherein the function calls can include

2   system calls or lib calls.

1   27.   The apparatus of claim 19, wherein the parameters can include:

2   file paths;

3   thread flags; and

4   timer-thread relationships.

14

Attorney Docket No. SUN-P6316-RSH   Inventor(s): Mathiske, et al.

ARPH·\SUN MICROSYSTEMS\SUN-P6316-RSH\SUN-P6316-RSH APPLICATION DOC